CLASSROOM INTEGRATION OF TECHNOLOGY: ARE TEACHERS UNDERSTANDING?

By

Dr. JERRY P. GALLOWAY

Georgia Southern University.

ABSTRACT

Teachers continue to be trained following a ritualized approach for skills and competencies. But, a deeper understanding of fundamental concepts, improvement of problem-solving and higher-order thinking skills and even the development of a contextual intuition can be even more important in becoming computer-using professionals. These factors as well as teachers' personal and professional adoption and personal commitment to a computing life-style can be critical prerequisites for achieving technological integration in tomorrow's school classrooms.

INTRODUCTION

Skills and competencies are the focus of the training teachers in instructional technology (AECT, 2007; ISTE, 2007). Showing teachers how to use a full range of state-of-the-art technologies make up the accepted curriculum of computing programs. But, of course, what constitutes state-of-the-art, changes rapidly and skills and competencies can become obsolete very fast. The training of teachers to use technology professionally and in teaching has attempted to keep pace with the quickly changing world of hardware innovation.

So, what are teachers actually learning? How can knowing how to use a given technology today which becomes obsolete tomorrow constitute being truly skilled or competent? Skill to use technology in the classroom is becoming more and more important and virtually all teachers are or will soon be expected to be successful integrators of technology in teaching (Wired News, 2007). In Vail, Arizona, Empire High School has done away with textbooks in favor of laptop computers. Of course, teachers are expected to not only to adapt but also to be the leaders in this evolution.

However, educators have a hard time with this challenge and continue to struggle to learn and develop skills in computing (Lipscomb & Doppen, 2005), with the evolution of hardware, software and computing techniques. The fundamental understanding of computing concepts have changed a lot. The educators are still having difficulties in computing, eventhough the schools are using computers for over 30 years.

A Historical Perspective

Perhaps a focus on understanding computing needs higher priority. There were some attempts made to understand computers and what they could be for us in learning and in society (Beaty & Tucker, 1987). But, the conceptual fundamentals of computing might have evolved from earlier years along with the hardware and other advancements (Galloway, 1990). Fundamentals at that time included notions of File, Data, Command, Program and Language. Even at that time the notion of a computer language was somewhat convoluted. Perhaps today, with the proliferation of user-friendly software, exposure to languages, per se, is rare and, in turn,perhaps the concept is even more elusive. Current research by Galloway is reexamining what concepts might considered fundamental in a variety of situations and looking at students' misconceptions in a modern context. Perhaps notions of input and output, variables, hard disk storage structures, telecommunications or various aspects of the graphical-user-interface would be more important and critical to understand computing.

There was an early division between those who learned to program computers versus those who learned little or nothing of that focusing more exclusively on applications software. Conceptual development, improvement of problem-solving and higher-order thinking skills in computing have been directly linked to the inclusion of Logo programming (Allen, 1993; Battista, 1994; Borer, 1993; Dalton & Goodrum, 1991) and BASIC programming (Overbaugh, 1993). Yet, in spite of an overwhelming need to operate early microcomputers through programming and to learn the fundamentals of computing in a broader sense, educators in particular resisted, focused instead on the actions and procedural tasks of specific applications.

A few dozen years of training have followed in which educators have continued to try to master the quickly changing world of technology. With the goals of mastering each popular product, computing procedure, each required task, educators have continued to react by quite literally showing teachers how to perform each task and procedure. So, how long does it take to reach a point of nation-wide competency, to develop the protocols of effective use, to establish the knowledge of how best to learn computing? The approaches across centuries of science and mathematics education seem quite different when compared to our 30-plus years of technology in classrooms. What it means to learn and specifically how people learn computing affects both what goals we pursue and how we proceed. For example, the use of rubrics or portfolios was not commonly emphasized in education 30 years ago. Today, however, they are an accepted or at least popular tools for preparing educators (Galloway, In-Press; The Rural School and Community Trust, 2001).

Training Teachers

The attempts taken to change educators into computer-using, computer literate professionals has not gone smoothly. Many will argue that there are countless success stories. But, with the exception of the techies and innovative pioneers, educators across the profession a generation ago did not or have not changed their basic approach to integrate technology. Compared to in-service classes, college courses, service center training or other options, an overwhelming majority of teachers maintain that their primary methods of learning computing was essentially by teaching themselves (Galloway, 1997). College-level computer education is commonly available for educators today but personal commitment and responsibility to learn computing is still critical.

It is a common notion that computing should be frustration-free (Simonsen & Gossman, 2007) and good teaching should somehow avoid anxiety.

Educators are often passive learners and restrict their involvement to occasional and discrete enrichment activities usually offered through someone else's initiative. Delays, intermittent involvement, partial commitments and other limitations inhibit learning. The computer using generation and its predecessors could be compared to the change in music trends over time and its acceptance and taste preferences among different generations of people.

Our efforts a generation ago were ineffective. We have simply waited around while a new generation, grew older bringing their technology-based lifestyle with them. Until our children have time to take their place, today's teachers are still introduced to computing as beginners.

Procedural Rituals

If we accept the notion that it is difficult to teach someone who doesn't want to learn, what do students expect from their training? Unfortunately, the most popular notion in instructional technology is that teachers are to be trained, not educated. More than mere semantics, teaching tends to emphasize *showing teachers how to use* technology - rather than facilitating insight, understanding, with a sound conceptual base. In-service programs and even college curricula emphasize only what teachers are expected to use rather than what might develop good concepts. Omitting programming is a classic example where teachers as end-users of software never see the construction process or design methods behind what they are supposed to learn. Today's design and preparation of web pages is a more modern example where these issues still apply.

Conceptual understanding and developing higher-order thinking skills should be a fundamental goal of instruction for beginners in computing, not just an effect of teaching programming (Tu & Falgout, 1995). Even with today's well-structured, modern standards for skills and competencies, students still fail to acquire important understandings, perspectives, concepts - integrated knowledge that all contribute a fundamental and critical basis for problem-solving and adaptability.

Naturally, focusing on conceptual development will involve procedures, tasks, keystrokes, software, projects, etc., just as focusing on mere training and discrete sets of tasks will likely yield some insights and discoveries. But, instruction should be designed to yield a more complete, sound and fundamental understanding of computing. Most programs - indeed, most perspectives of teacher educators - fail to recognize this important viewpoint and instead pursue skills and competencies to the detriment of understanding, insight and problem solving.

It is common in other discipline areas to speak of *education* rather than *training.* Conceptual development is often the primary focus in the study of science (Trumper, 1997). Even when the preparation of teachers is described in terms of training, science concepts are emphasized, not skills (Thompson & Schumacher, 1995), in spite of the procedures and skills inherent in science experimentation and discovery and in mathematics, students are guided toward the development of a conceptual understanding as they are educated - not trained.

A training model targets activities, tasks, skills and what software teachers will use. An education model, on the other hand, calls for activities and experiences that will yield a deeper kind of learning. Certainly, keystrokes and software familiarity would be included but would also be incidental to the more important yield of experiences, much like those in science and mathematics, that develop understanding, a strong conceptual base, problem solving and critical thinking skills. An education-based program would provide such experiences because of their educational value regardless of the existence of specific software or procedures in an anticipated skill set. Skill sets, tasks and the procedural rituals of training will inevitably change and evolve far beyond the scope of any training experience.

Student teachers can be part of the problem as they, very often, prefer the training model. Contrary to any real value or longevity of such an approach, a more involved education presents students with an undesirable challenge. They prefer to simply be shown what to do. Guided tasks, prescribed procedures, discrete tasks are all a matter of things that can be done, not things that can be achieved. This works well for training. An education, on the other hand, calls for change.

Acquiring mindless task sequences is often viewed by educators as success. Improved teaching is then viewed as having more complete check lists for more and more tasks. This *recipe mentality* of discrete procedural rituals ignores the need for discovery learning, transfer and adaptability and could be responsible for continuing the inhibited progress of integrating computing into our professional lives.

Into the School Classroom

As Galloway (1997) further examined the adoption of technology by teachers, it was learned that effective usage is related to the combination of both professional and personal adoption of computer technology. Virtually no one used technology in their classrooms where personal adoption was not combined with professional use. Additional factors for the integration of technology include funding, professional development, support for experimentation, and inadequate technology planning (Mehlinger & Powers, 2002).

It has been said that teachers exist only for the children. They express the sentiment that the student needs are the primary if not the only mission of teachers. It is easy, however, to draw the wrong conclusions from such a self-evident premise. For instructional technology, consistent with this perspective, trends have been directed away from empowering teachers directly to focus instead on classroom integration. This may seem justified but a serious problem remains: it is not reasonable to teach non-computer-users to use technology in the classroom. Educating teachers to become computer-using, technology-competent professionals would more likely yield classroom integration as a matter of natural consequence.

While there are training programs targeting classroom integration, what success can be had if teachers are not computer-literate or have never adopted computing in their lives? In other words, one must adopt technology as a life-changing metamorphosis. The approach of the past i.e., training over education, rituals and isolation over holistic adoption may likely to continue the inhibited progress toward integration and technological mastery.

Teachers continue to be trained in accordance with the mindset that a computer is a tool (Beaty & Tucker, 1987). An alternative view might suggest that the computer is not a tool at all. As a tool might be selected or discarded based on a particular need, technology is too often viewed as independent from everyday life, powered-up if the need is sufficiently demanding. Instead, computers are perhaps best viewed as a complete environment. It is presented where we live, work and play. It is the medium of our planning, our creativity and an extension of both our short-term and long-term memory. This thinking places different expectations on educators than has traditionally been made. The notion that one can remain a non-computer-using person while merely executing discrete tasks as needed must change.

Summary

Learning is no longer about the acquisition of information. Acquisition of information is neither the problem nor the goal because information is readily available both traditionally and electronically throughout our society. Students assume that they are to acquire facts and they resist anything more personally demanding. The notion that they must change, must invent or synthesize is foreign to them. Learning to think is the real challenge. Education has become art of skill development with demonstrable competencies rather than becoming smarter or learning to think. Education, as distinct from training, should improve problem solving abilities, critical thinking abilities, developing an understanding, learning to discriminate and make good choices, and the development of a contextual intuition.

Computing students do not want to have to explore and discover. Instead, they want to be shown how to execute procedures. Being ready for tomorrow's computing world depends on a strong conceptual base, problem solving skills and the ability to adapt the unknown.

Software and hardware continue to develop faster than anyone can sufficiently learn them. Merely being able to operate the functions and tools in each package is usually considered a success. But, achieving a deeper knowledge of how best to adopt, integrate and teach in a world of technology is quite a different thing. It might be argued that repeating the mindset and methodologies of the past will not improve our future.

The overwhelming conclusion of the past 30 years is that training for discrete tasks must be replaced by holistic adoption and education. Being successful at computing is not a function of memorized procedures or specific skill sets. Procedural rituals, however conveniently arranged or exhaustively accounted, cannot substitute for intuition, problem solving and a deeper understanding of computing. Successful computing depends directly on such deeper learning and the personal adoption of computing by teachers both personally and professionally.

References

[1]. **AECT (2007).** *Association for Educational Communications and Technology Initial Standards.* Retrieved February 6, 2007, from http://www.aect-members.org/standards/initstand.html

[2]. **Allen, J. (1993).** The impact of cognitive styles on the problem-solving strategies used by preschool minority children in Logo microworlds. *Journal of Computing in Childhood Education, 4* (3-4), 205-217.

[3]. **Battista, M. T. (1994).** Research into practice: Calculators and computers: Tools for mathematical exploration and empowerment. *Arithmetic Teacher, 41* (7), 412-417.

[4]. **Beaty, J. J., & Tucker, W. H. (1987).** *The computer as a paintbrush.* Columbus, OH: Merrill Publishing Co.

[5]. **Borer, M., (1993).** *Integrating mandated Logo computer instruction into the second grade curriculum.*

M.S. Practicum Report, Nova University (ERIC Document No Ed367311).

[6]. **Dalton, D. W., Goodrum, D. A., (1991).** The effects of computer programming on problem-solving skills and attitudes. *Journal of Educational Computing Research, 7* (4), 483-506.

[7]. **Galloway, J. P.** (In-Press). Electronic Portfolios for Educators. *International Journal of Arts and Sciences.*

[8]. **Galloway, J. P. (1990).** Misconceptions of computing concepts among preservice teachers. *Journal of Research on Computing in Education, 22* (4), 413-430.

[9]. **Galloway, J. P. (1997).** How Teachers Use and Learn To Use Computers. *Technology and Teacher Education Annual Journal, 1997.* Published by the Association for the Advancement of Computing in Education, Charlottesville, VA.

[10]. ISTE (2007). *ISTE / NCATE Standards for Educational Technology Programs.* Retrieved February 6, 2007, from http://cnets.iste.org/ncate/

[11]. **Lipscomb, G. B., & Doppen, F. H. (2005).** Climbing the stairs: Pre-service social studies teachers' perceptions of technology integration. *International Journal of Social Education, 19,* 2.

[12]. **Mehlinger, H. D., & Powers, S. M. (2002).** *Technology and teacher education: A guide for educators and policymakers.* Boston, Mass.: Houghton Mifflin.

[13]. **Overbaugh, R. C., (1993).** *A BASIC programming curriculum for enhancing problem-solving ability.* Evaluative Report, Darden College of Education, Old Dominion University, Norfolk, Virginia. (ERIC Document No Ed355921).

[14]. **Simonsen, J. & Gossman, S. (2007).** *Curing computer frustration.* Retrieved February 6, 2007, from http://www.isd77.k12.mn.us/~jsimon1/intro.htm

[15]. The Rural School and Community Trust (2001). *Assessing student work.* Retrieved January 8, 2007 from: http://www.ruraledu.org/site/c.beJMIZOCIrH/b.1389103/apps/s/content.asp?ct=838177

[16]. **Thompson, G. W. & Schumacher, L. G., (1995).** Implications of integrating science in secondary agricultural education programs. *Paper presented at the American Vocational Association Convention* (Las Vegas, NV, December 1997).

[17]. **Trumper, R. (1997).** The need for change in elementary school teacher training: The case of the energy concept as an example. *Educational Research, 39* (2), 157-74.

[18]. **Tu, Jho-Ju, & Falgout, B. (1995).** Teaching If-Then Structures: An Integrated Approach. *Learning and Leading with Technology, 23* (3), 26-28.

[19]. **Wired News (2007).** *Look, Ma, No Schoolbooks!* Retrieved February 6, 2007, from http://www.wired.com/news/technology/0,1282,68578,00.html?tw=wn_tophead_6

ABOUT THE AUTHOR

Dr. Jerry Galloway, obtained his bachelors degree in Logic and Law at the South Texas College of Law, and Doctor of Education degree from the University of Houston in 1987.

After a couple of years as professor at the University of Houston, Dr. Galloway served for 16 years as the Coordinator of Computer Education and Associate Professor of Education at Indiana University Northwest. He has published dozens of articles, a number of textbooks and a variety of CD media materials. He has now joined the instructional technology faculty at Georgia Southern University.